>CURSOR™

THE TINY MICRO COMPUTER NEWS SERVICE

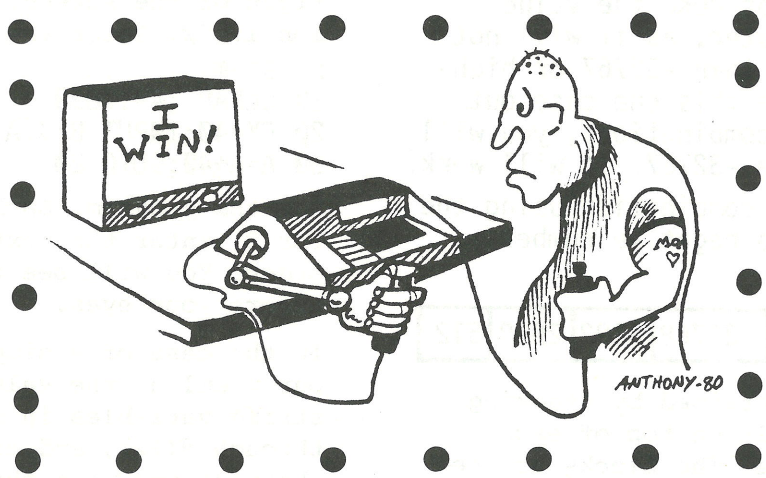
VOLUMN I $2.00 per issue APRIL/MAY 1980 $9.50 Semi-Annually ISSUE 4

GOAL OF CURSOR

Why have you subscribed to CURSOR? The CURSOR Staff hopes that your reasons are the same as ours in producing CURSOR, i.e. To have the ability to create programs as complex as those BALLY has produced. In our March issue we gave you the "Three Voice Music Assembler". That issue gave you the ability to break out of one voice "Kiddee" melodies and play 3 Voice Bach Cantatas"!

This issue will do the same for "Graphics. Your graphics capability is now limited only by your imagination and the amount of graph paper on hand.

NOTE: DMA is an abbreviation of Direct Memory Access. It uses our old friend "PEEK n' POKE" described in our February issue.



DMA GRAPHICS
BY
C.J. ANDERSON

The Bally screen display is composed of 14,080 pixels. A pixel is the smallest possible dot, the size of a period, that the Bally Computer can display, measuring one unit wide by one unit high. All of Bally's graphics, including letters and words, are made up of these pixels.

The dimensions of the screen are 160 pixels wide by 88 pixels high. There was an error in the early Bally Programming Course booklet (page 64, Lesson 6: Graphics) where the screen dimensions were given as 159 dots wide and 87 dots high. The author forgot to count the center lines, X=Ø & Y=Ø, in his measurements. The pixels are numbered from -80 on the far left to 79 on the far right, and from -44 at the bottom to 43 at the top. Counting the center lines at coordinate Ø gives us the correct dimensions of 16Ø by 88.

Each pixel has two possible states. It may be the same color as the background, or "off." We will call this state"Ø". On the other hand it may be the same color as the foreground, or "on." We will call this state "1."

The pixels are grouped into horizontal blocks of eight, making the screen 20 blocks wide by 88 blocks high. These blocks of eight pixels each are controlled by the even-numbered screen memory locations from 16384 (upper-left-hand corner) to 19902 (lower right hand corner). By "poking" any one of 256 possible values into an even-numbered screen memory location you can turn the eight pixels controlled by that location "on" and "off" in every possible combination. This technique, called DMA, allows you to include intricate details within graphics without using as much memory as you would need to do the same thing using the BOX command in Bally Basic.

There are two ways to determine the value of a particular block pattern of eight pixels. The simplest is to write the desired pattern on a piece of paper as an eight digit binary number, using "1" for each pixel you want "on" and "Ø" for each pixel you want "off." Once you have the number

MAIL ADDRESS: P.O. BOX 266, NORTH HOLLYWOOD, CA 91603
BUSINESS ADDRESS: 6115 CLYBOURN, SUITE 25, NORTH HOLLYWOOD, CA 91606 • (213) 763-7701

simply look it up in the provided table. You will find the equivalent value there.

EXAMPLE: Suppose you want to turn on evevy other pixel in the block located in the exact center of the screen. That block is controlled by screen memory location 18124. Write down the binary number 1Ø1Ø1Ø1Ø and look it up in the table. The equivalent value is given as -30584. Clear the screen, and enter the following command: %(18124)=-3Ø584. There's your dotted line in the center of the screen.

The second method is to calculate the values yourself using the following diagram. Simply add together the numbers contained in each box (representing a pixel) that you wish to turn on. Our previous example could be obtained by adding every other number: 128 plus 8 plus -32768 plus 2Ø48. The answer will be the same: -3Ø584.

Notice the value of the fifth pixel: -32 768. Suppose you wanted to turn on that pixel only. You cannot POKE the value -32768 into the computer, as it will not accept numbers lower than -32767 or higher than 32767. So, in this one case out of the 256 possible combinations, you will have to use the value -32767. It will work.

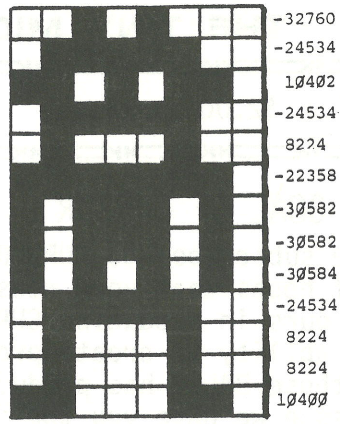
Notice also that any combination using the fifth pixel will be a negative number.

128	32	8	2	-32768	8192	2Ø48	512

Fancy graphics are obtained by stacking blocks of eight pixels on top of each other or side-by-side. The blocks are separated by 40 vertically and by 2 horizontally. For example, the block immediately below the block in the center of the screen, 18124, is located at memory location 18164. The block immediately above is located at memory location 18Ø84. The block to the immediate left is at 18122 and the block to the immediate right is at 18126. You can make your own screen memory map by ruling off a sheet of paper into 20 columns and 88 rows. Label each box by 2's, beginning with 16384 in the upper left and proceeding horizontally: 16386, 16388, 1639Ø etc. until you reach 199Ø2 at the lower right. Each box represents eight pixels.

For a simple example, suppose you want a little gremlin to use in a video game you're designing. First draw him, using multiples of eight pixels horizontally. We'll begin with a gremlin only one block wide.

The gremlin is 13 blocks high. Using the table or the diagram on the opposite side of this page, find the 13 values needed to draw this little fellow.



Now use your screen memory map to decide where you want the gremlin to appear. As an example, let's put him in the center right of the screen. We'll start at location 1789Ø. Enter the following simple program:

```
1Ø CLEAR ;A=1789Ø
2Ø CY=4Ø;INPUT B;%(A)=B
3Ø A=A+4Ø;GOTO 2Ø
```

RUN the program. Each time the screen asks for B, enter the next of the 13 values in order. You will see the gremlin take shape before your eyes.

In the case of a simple graphic such as our gremlin, the values can be stored as string variables in string locations @(Ø) through @(12), and accessed by a subroutine whenever you want the gremlin to appear. You can place him anywhere on the screen simply by changing the address of variable A.

EDITOR'S NOTE:
Our computer stores your basic program on screen starting at the top and working down. If you POKE graphics into a screen location already occupied by a Basic program, you will eat up the program. You can find out how much screen the BASIC program is occupying by setting &(9)=Ø plus &(Ø) - &(1) - &(2) - &(3) to different color values. The garbage at the top of the screen is your Basic program. To return screen to normal, set &(9)=5Ø.

If you need to place graphics into an area already occupied by your Basic program, it can be done by adding the value already stored in the screen location to the value

you wish to POKE. EXAMPLE: If we wanted to POKE location 16386 with 8224, it would be stated thusly:

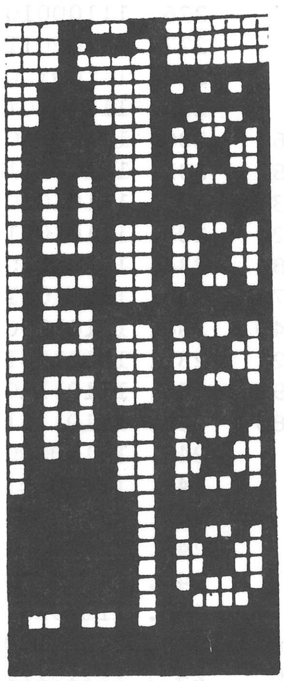
%(16386)=8224

If we have a program stored, this location is already occupied and after poking you will have destroyed the program. To get it to work, the statement must read:

%(16386)=%(16386)+8224

This is a tad more cumbersome, but it works!!!

While a gremlin takes very little memory (imagine drawing him using BOX commands!), a more complicated graphic such as the spaceship (later in article) with its launching gantry, two blocks wide by 40 blocks high, can be a real memory-eater. You will need to POKE 80 values to draw this rocket while using two lettered variables to keep track of the screen memory locations. Storing these values as string variables will require 160 bytes of memory. If the spaceship is the beginning of a game or an animated cartoon, you might consider storing these values directly on tape and loading them directly into screen memory from tape, bypassing the text memory area. That will save 160 bytes that you can use for your program.



1.	-32254	2.	128	3.	1Ø242	4.	128
5.	2	6.	128	7.	2	8.	-21846
9.	-32758	1Ø.	8866	11.	-32758	12.	-21846
13.	-32758	14.	272Ø	15.	-24534	16.	884Ø
17.	-24534	18.	-32126	19.	-24534	2Ø.	-32126
21.	8226	22.	884Ø	23.	8226	24.	-21846
25.	1Ø786	26.	-21846	27.	8226	28.	884Ø
29.	8224	3Ø.	-32126	31.	-24534	32.	-32126
33.	8224	34.	884Ø	35.	-24542	36.	-21846
37.	1Ø784	38.	-21846	39.	8234	4Ø.	884Ø
41.	8224	42.	-32126	43.	-24534	44.	-32126
45.	8224	46.	884Ø	47.	8226	48.	-21846
49.	1Ø784	5Ø.	-21846	51.	8226	52.	884Ø
53.	8226	54.	-32126	55.	-24534	56.	-32126
57.	-24534	58.	884Ø	59.	-22358	6Ø.	-21846
61.	-22358	62.	-21846	63.	-22358	64.	884Ø
65.	-22358	66.	-32126	67.	-22358	68.	-32126
69.	-22358	7Ø.	884Ø	71.	-22358	72.	272Ø
73.	2178	74.	-21846	75.	-21846	76.	-21846
77.	-21846	78.	-21846	79.	-21846	8Ø.	-21846

Loading this puppy is a little complicated the first time, so let's take it carefully, step-by-step. Snap a blank tape into your cassette recorder, but don't start recording yet.

STEP ONE: Program the computer to clear the screen, open the input port and arrange the incoming data on the screen in a 2 x 40 block pattern at the bottom center of the screen, beginning with the rocket's nose at location 18284. Enter this program:

```
1Ø CLEAR ;:INPUT ;FOR A=18284 TO 19844STEP 4Ø
2Ø CY=4Ø;INPUT B;%(A)=B
3Ø CY=4Ø;INPUT B;%(A+2)=B
4Ø NEXT A;STOP
```

STEP TWO: Load that program onto cassette tape using a self-starting command:

```
NT=Ø;:PRINT ;LIST ;PRINT ":RETURN ;TV=13;RUN
```

As soon as the program is loaded on tape, press the PAUSE button on your recorder. (If you don't have a PAUSE button, stop the recorder. Do not rewind!)

STEP THREE: RESET the computer. Now program the computer to accept the 80 graphic values and store them as consecutive string variables. Enter this program:

```
1Ø CLEAR ;FOR A=1TO 8Ø
2Ø INPUT B;@(A)=B;NEXT A
```

RUN the program. The computer will ask for B.Enter the first value: -32254. Press GO. The computer will again ask for B. Enter the second value: 128. Continue to enter values in that manner (third value:1Ø242, fourth value: 128, etc.) until all 80 values are stored.

STEP FOUR: Load those values onto cassette tape using the following direct command:

```
NT=Ø;:PRINT ;FOR A=1TO 8Ø;CY=4Ø;PRINT @(A);
NEXT A
```

0.	00000000	0
1.	00000001	512
2.	00000010	2048
3.	00000011	2560
4.	00000100	8192
5.	00000101	8704
6.	00000110	10240
7.	00000111	10752
8.	00001000	-32767
9.	00001001	-32256
10.	00001010	-32720
11.	00001011	-30208
12.	00001100	-24576
13.	00001101	-24064
14.	00001110	-22528
15.	00001111	-22016
16.	00010000	2
17.	00010001	514
18.	00010010	2050
19.	00010011	2562
20.	00010100	8194
21.	00010101	8706
22.	00010110	10242
23.	00010111	10754
24.	00011000	-32766
25.	00011001	-32254
26.	00011010	-32718
27.	00011011	-32206
28.	00011100	-24574
29.	00011101	-24062
30.	00011110	-22526
31.	00011111	-22014
32.	00100000	8
33.	00100001	520
34.	00100010	2056
35.	00100011	2568
36.	00100100	8200
37.	00100101	8712
38.	00100110	10248
39.	00100111	10760
40.	00101000	-32760
41.	00101001	-32248
42.	00101010	-30712
43.	00101011	-30200
44.	00101100	-24568
45.	00101101	-24056
46.	00101110	-22520
47.	00101111	-22008
48.	00110000	10
49.	00110001	522
50.	00110010	2058
51.	00110011	2570
52.	00110100	8202
53.	00110101	8714
54.	00110110	10250
55.	00110111	10762
56.	00111000	-32758
57.	00111001	-32246
58.	00111010	-30710
59.	00111011	-30198
60.	00111100	-24566
61.	00111101	-24054
62.	00111110	-22518
63.	00111111	-22006
64.	01000000	32
65.	01000001	544
66.	01000010	2080
67.	01000011	2592
68.	01000100	8224
69.	01000101	8736
70.	01000110	10272
71.	01000111	10784
72.	01001000	-32736
73.	01001001	-32224
74.	01001010	-30688
75.	01001011	-30176
76.	01001100	-24544
77.	01001101	-24032
78.	01001110	-22496
79.	01001111	-21984
80.	01010000	34
81.	01010001	546
82.	01010010	2082
83.	01010011	2594
84.	01010100	8226
85.	01010101	8738
86.	01010110	10274
87.	01010111	10786
88.	01011000	-32734
89.	01011001	-32222
90.	01011010	-30686
91.	01011011	-30174
92.	01011100	-24542
93.	01011101	-24030
94.	01011110	-22494
95.	01011111	-21982
96.	01100000	40
97.	01100001	552
98.	01100010	2088
99.	01100011	2600
100.	01100100	8232
101.	01100101	8744
102.	01100110	10280
103.	01100111	10792
104.	01101000	-32728
105.	01101001	-32216
106.	01101010	-30680
107.	01101011	-30168
108.	01101100	-24536
109.	01101101	-24024
110.	01101110	-22488
111.	01101111	-21976
112.	01110000	42
113.	01110001	554
114.	01110010	2090
115.	01110011	2602
116.	01110100	8234
117.	01110101	8746
118.	01110110	10282
119.	01110111	10794
120.	01111000	-32726
121.	01111001	-32214
122.	01111010	-30678
123.	01111011	-30166
124.	01111100	-24534
125.	01111101	-24022
126.	01111110	-22486
127.	01111111	-21974
128.	10000000	128
129.	10000001	640
130.	10000010	2176
131.	10000011	2688
132.	10000100	8320
133.	10000101	8832
134.	10000110	10368
135.	10000111	10880
136.	10001000	-32640
137.	10001001	-32128
138.	10001010	-30592
139.	10001011	-30080
140.	10001100	-24448
141.	10001101	-23936
142.	10001110	-22400
143.	10001111	-21888
144.	10010000	130
145.	10010001	642
146.	10010010	2178
147.	10010011	2690
148.	10010100	8322
149.	10010101	8834
150.	10010110	10370
151.	10010111	10882
152.	10011000	-32638
153.	10011001	-32126
154.	10011010	-30590
155.	10011011	-30078
156.	10011100	-24446
157.	10011101	-23934
158.	10011110	-22398
159.	10011111	-21886
160.	10100000	136
161.	10100001	648
162.	10100010	2148
163.	10100011	2696
164.	10100100	8328
165.	10100101	8840
166.	10100110	10376
167.	10100111	10888
168.	10101000	-32632
169.	10101001	-32120
170.	10101010	-30584
171.	10101011	-30072
172.	10101100	-24440
173.	10101101	-23928
174.	10101110	-22392
175.	10101111	-21880
176.	10110000	138
177.	10110001	650
178.	10110010	2186
179.	10110011	2698
180.	10110100	8330
181.	10110101	8842
182.	10110110	10378
183.	10110111	10890
184.	10111000	-32630
185.	10111001	-32118
186.	10111010	-30582
187.	10111011	-30070
188.	10111100	-24438
189.	10111101	-23926
190.	10111110	-22390
191.	10111111	-21878
192.	11000000	160
193.	11000001	672
194.	11000010	2208
195.	11000011	2720
196.	11000100	8352
197.	11000101	8864
198.	11000110	10400
199.	11000111	10912
200.	11001000	-32608
201.	11001001	-32096
202.	11001010	-30560
203.	11001011	-30048
204.	11001100	-24416
205.	11001101	-23904
206.	11001110	-22368
207.	11001111	-21856
208.	11010000	162
209.	11010001	674
210.	11010010	2210
211.	11010011	2722
212.	11010100	8354
213.	11010101	8866
214.	11010110	10402
215.	11010111	10914
216.	11011000	-32606
217.	11011001	-32094
218.	11011010	-30558
219.	11011011	-30046
220.	11011100	-24414
221.	11011101	-23902
222.	11011110	-22366
223.	11011111	-21854
224.	11100000	168
225.	11100001	680
226.	11100010	2216
227.	11100011	2728
228.	11100100	8360
229.	11100101	8872
230.	11100110	10408
231.	11100111	10920
232.	11101000	-32600
233.	11101001	-32088
234.	11101010	-30552
235.	11101011	-30040
236.	11101100	-24408
237.	11101101	-23896
238.	11101110	-22360
239.	11101111	-21848
240.	11110000	170
241.	11110001	682
242.	11110010	2218
243.	11110011	2730
244.	11110100	8362
245.	11110101	8874
246.	11110110	10410
247.	11110111	10922
248.	11111000	-32598
249.	11111001	-32086
250.	11111010	-30550
251.	11111011	-30038
252.	11111100	-24406
253.	11111101	-23894
254.	11111110	-22358
255.	11111111	-21846

Release the PAUSE button on the recorder (or begin recording) then press GO. The 80 values will flash across the top of the screen as they load onto the tape. When the last value has loaded, type :RET URN, then stop the tape. You're done!

RESET the computer and rewind the tape. Now type :INPUT and play back the tape. You will see the first program load, then the screen will clear. Now, as the 80 values flash across the top of the screen, the spacecraft and its gantry will appear at the bottom center. Stop the recorder; there's your graphic!

It may have occurred to you that it is possible to write a half-hour long animated cartoon and play it on the screen directly from the cassette tape, bypassing computer memory entirely. Sounds like a lot of work, but for a trade show or a business presentation it would certainly be an attention-getter. Why, you could even get paid for it!!

Gain experience by designing your own graphics. The simplest way is to use a sheet of 160 by 88 graph paper. Draw your graphic using normal lines and shading, then go back and black in every square that is crossed by a line or is within a shaded area. Break the final picture into eight pixel blocks, encode them and put them on the screen. You can do quite detailed pictures, even photos, if you wish. A full screen picture of a face, viewed from a distance, takes on a heavily-screened newspaper picture quality.

Animation involves the blanking out of a block and either moving the entire block (rough animation) or rearranging the pixels within the block (fine animation). The spaceship you just drew has been animated as follows: the umbilical cord drops away from the nose of the rocket, then the walkways retract into the gantry. The entire gantry then rolls away from the rocket, off the right-hand side of the screen. A countdown begins. At zero, smoke and flames rise from the launch platform and the rocket majestically rises and disappears off the top of the screen. (In another version the rocket rises only half-way up the screen, then slowly sinks back down and crumbles into a pile of rubble at the bottom of the screen. We call that one VANGUARD ONE!) Go ahead and try your hand at animating the rocket; it's a good practise subject. You'll need graph paper and pocket calculator as you move the pixels around. Your work may seem slow, but the computer will process the movement of the pixels so rapidly that the eye will perceive it as a smooth movement. In some cases you may have to slow things down to make them appear more realistic (we made the umbilical cord bounce several times against the side of the gantry for added realism.)

Other valuable uses for DMA graphics include type fonts. Now you can spice up your charts and graphs with type faces ranging from lower-case serif and sans serif to Old English or Spencerian script. Or design your own type style.

If you get really serious about DMA graphics, we'd enjoy seeing samples of your work. But please: no more Snoopy calendars!

"A COMPUTER IN THE HANDS OF AN ARTIST IS AS VALID A TOOL AS A PAINT BRUSH OR CHISEL."

Flash Foonman

NOTES, NOTICES & NODS

We have had numerous requests for info regarding BACK ISSUES! There are 3 available:

1-Jan 80 Contains: Electric Bill Analysis; Plastic Puzzle; Instructions for adding a Full sized ASCII Keyboard; Life Synthesis Model.

2-Feb 80 Contains: PEEK n' POKE; Hex to Decimal Converter; String Array @(A) Memory Locator; Instructions on how to add a Printer; Bubble Sort; Camel; Memory Map; WUMPUS.

3-Mar 80 Contains: Three Voice Music Assembler; Star Wars Music; Chopsticks; Chicago Loop; Lace Curtain; Technical Manuals; Character Set Size Multiplier; Rotation; National Distributor Info.

All back issues are available to subscribers for $1.60 each. Please specify issues desired.

A clarification has been asked for, regarding technical manuals offered in the March Issue.

#1-On Board Sub-Routines: allow you to perform complex operations such as the "Character Set Size Multiplier" and many many more that are described no where else.

#3-Disassembled Tiny Basic: An Assembly Language Listing (complete with Object

Code and Comments) of the Tiny Basic Cartridge. This manual allows you to understand how the Tiny Basic works and will allow you to perform higher level operations.

All of the manuals we offered in our March Issue are priced without profit (we don't have access to a free Xerox machine at work!)

No doubt you have noticed the date of this issue (April/May)! Our rapid growth has created many problems for us. Our ability to handle the volumn of orders and correspondence has not grown proportionately. We have now added additional office staff and have decided it would take several months to catch up and be able to get an issue out on time. To eliminate this irritating effect on our subscribers we have made this issue a one-time two month issue. You will receive the June issue by the 10th of June; and every issue thereafter by the 10th of the month. NOTE: A subscription is six issues not necessarily six months- you will receive six issues.

BOOK ORDERS

If you have had difficulty finding the books we recommend within our pages, we have made arrangements for you to order them in person, by mail, or phone (and also receive a 10% Discount). Available are:

1. TV TYPEWRITER COOKBOOK by Don Lancaster
2. THE BASIC HANDBOOK by David A. Lien.

Please call or write for prices (mention CURSOR for discount).

OPAMP/
Technical Books
1033 N. Sycamore Ave.
Los Angeles, CA. 90038
(213) 464-4322
ATTN: Lyn Karch

THE COMPUTER EAR

A PRODUCT REVIEW

BY

FRED CORNETT, EDITOR

Have you, as I have, dreamt of controlling games and devices by voice instead of hand control?

You can imagine my delight when opening the mail recently to find a "SPEECH RECOGNITION SYSTEM" custom made for the Bally Arcade!! (This type of device is selling for upwards of $195. for other computers.

I immediately dropped everything and rushed over to my computer to connect it and try it out. Boy, was that easy!! It plugs into Hand Control Port #4, and requires a cheap 9 volt transistor radio battery.

The "Computer Ear" comes with a very detailed 20 page User Manual and includes:

1. Audio amplifier with optically isolated analog data output.
2. Power cord with DC plug & battery clip.
3. Computer input cable with RS-232 9-pin female connector.
4. USER Manual.
5. Program Cassette (4 programs).

After receiving a mini-education in SPEECH RECOGNITION from the manual, we loaded the first program "Digital Oscilloscope" which allows you to freeze words on the screen and compare waveforms against one another as an aid to vocabulary selection.

WOW!!! This product really works! This device allows you to select the complexity level you desire from comparing whole sentences to simple one word amplitude averaging, such as "UP"versus "DOWN", LEFT vs RIGHT, or YES vs NO. Amplitude averaging is the easiest to use and requires almost no more memory than a simple hand control statement.

One thing that must be remembered is that the "Computer Ear" is not using ASCII code as a printer would, leaving no room for error. The "Ear" is as exact as your voice is constant. In other words, you will have to use the "Digital Oscilloscope" a great deal in the beginning until you learn how to say a word the same way every time you use it (accuracy potential is pegged at over 80%).

We wrote a couple of game programs, and shortly thereafter were moving our playing pieces around by voice. The potential on the "EAR" is unlimited.

The most amazing thing about the "Computer Ear" is its' price; $59.95 COMPLETE (Minn. residents add 4% sales tax).

If you would like to have a peripheral device that will occupy a few months of your time and amaze your non-computer/fanatic friends, the "Computer Ear" is the answer. This device is brought to you by the same people that wrote the "DMA Graphics" tutorial.

NOTE: CURSOR will support this device with software in future issues.

MUSIC CONTEST

Well, we have finally done it! LINE 21Ø of the "Three Voice Music Assembler"(Copyright 1980) contains a glitch. LINE 21Ø should read:

```
21Ø FOR C=ATO A+92STEP 4
```

We have had many, many requests to extend the time limit for the "Music Contest", so, bowing to popular demand, we hereby extend the time cutoff to JUNE 15, 1980.

Many thanks to Albert Paul of Winnipeg, Manitoba, Canada for the following program. NOTE: To make the Ring thicker or thinner, change the value of "N" (Line 2Ø) To make the Ring larger or smaller,change the value of "M" (Line 2Ø). "M" must always be larger than "N".

RING
BY
ALBERT PAUL

```
 1Ø CLEAR
 2Ø N=16ØØ;M=1849
 3Ø FOR X=-44TO Ø
 4Ø FOR Y=-44TO Ø
 5Ø IF (XxX)+(YxY)>MGOTO 11Ø
 6Ø IF (XxX)+(YxY)<NGOTO 12Ø
 7Ø BOX X,Y,1,1,1
 8Ø BOX X,-Y,1,1,1
 9Ø BOX -X,Y,1,1,1
1ØØ BOX -X,-Y,1,1,1
11Ø NEXT Y
12Ø NEXT X
```

ALARM CLOCK
BY
MIKE PEACE

```
  5 NT=Ø
 1Ø CLEAR ;INPUT "ALARM HOUR"A
 2Ø INPUT "ALARM MINUTE"L
 3Ø INPUT "HOUR"H
 4Ø INPUT "MINUTE"M
 5Ø INPUT "SECONDS"S;CLEAR
 55 CY=1Ø;CX=-37;PRINT "BALLY TIME
 6Ø S=S+1;M=M+I;H=H+R;I=Ø;R=Ø
 7Ø IF S=6ØI=1;S=Ø
 8Ø X=26Ø
 9Ø IF M=6ØR=1;M=Ø
1ØØ CY=Ø;IF H=13H=1
11Ø PRINT #1Ø,H,":",#1,M,":",#1,S
12Ø FOR T=1TO X;NEXT T
13Ø IF A=HIF L=M&(17)=2Ø;&(22)=255
135 &(2Ø)=4Ø
14Ø IF M#L&(22)=Ø
15Ø GOTO 6Ø
16Ø .M-PEACE
```

This clock runs very accurately over a prolonged period of time. The clock runs faster when the hour is a small number, and runs slower when the hour is a large number; these factors tend to equal each other out. If you have found the clock needs a little fine tuning, you can adjust the speed by changing the value of "X" in Line 8Ø(smaller number runs faster). If you make any changes in the program for Line 6Ø through 15Ø, you will throw off all timing.

NOTE: Mike Peace has published several programs in CURSOR, and is part of "WAVE MAKERS". Wavemakers recently sent CURSOR a copy of their software catalog and copies of their tapes. We were quite impressed with the diversity of their programs plus low prices,and find we can't seem to leave their version of "CLUE" alone. For those looking for a reliable software source, try investing 15¢ and send for their software catalog.

WAVE MAKERS
P.O. Box 94801
Shaumburg, IL 60193

BYTE-SAVING HINTS

Many programmers place unnecessary spaces within their statements, i.e., IF A=4 B=3. As long as the two characters you want to place together are not of the same type, you can do it, i.e., IF A=4B=3 or IF A=4GOTO 1Ø or IF A=BGOTO 1Ø;but <u>not</u> IF A=BC=4.Try this, it can save you quite a few bytes.

C L A S S I F I E D S

CURSOR SOFTWARE TAPE #1

The following two photographs and descriptions make up the first CURSOR Cassette Tape offering. Both programs are listed on one tape, and include complete documentation. Price is $8.95 complete. Send checks or money orders to: CURSOR, P.O. Box 266, N. Hollywood, CA 91603.

PROGRAM 1: MOON LANDING

You're in the L.E.M. waiting for the instruction to break away from the mother ship. Once you do, you have to quickly scout for a safe landing spot. You carefully maneuver into a safe landing position; watching the drift and speed. If you successfully land, you have to wait for the countdown to blast-off for re-connection to the mother ship prior to running out of time and fuel.(Software selectable gravity wells.) Program is partly in machine language to generate the fast acting user-defined characters for: Horizontal LEM, LEM banked to the right, LEM banked to the left, 2 explosions (moving). Great sound & graphics.

PROGRAM 2; BIO-RHYTHM

Through this computerized study of biological clocks you can predict physical, emotional and intellectual behavior at peak and critical periods. Bio-rhythm has helped U.S. airlines avoid crashes and athletes to choose their best competive days; it has reduced dramatically the auto accident rate in Japan and increased the performances of sales forces, teachers, and factory workers. Bio-rhythm can help you predict outbreaks of illness, mental depression, days of tireless energy, best times for creative work, peak periods of mental and emotional control.

Very accurate graph format allows you to select and see your critical days individually. No other program like it!

USER GROUP MEETING

Los Angeles Area User Group Meeting:
TIME: Wednesday, 21 May 1980 7:30 PM
PLACE: 5640 Fair Avenue, Apt. 21
North Hollywood, CA 91601
Phone: 213-763-0734

NEW SOFTWARE: If you enjoyed "Rotation" in our March Issue we suggest you buy a copy of "Super-Rotation" which is 10 times as complex and offers many variations; Bob spent several months cranking this one out. $5.50 postpaid to:
Robert Leake
297 S. Marengo, #309
Pasadena, CA 91101

CURSOR
P.O. BOX 266, NO. HOLLYWOOD, CA 91603

FIRST CLASS